Article

A systematic evaluation and benchmarking of summarization methods for biomedical literature: From word-frequency methods to language models

Fabio Baumgärtel [1,†], Enrico Bono [1,2,†], Lucas Fillinger [1], Louiza Galou [1], Kinga Keska-Izworska [1], Samuel Walter [1], Peter Andorfer [1], Klaus Kratochwill [1,2], Paul Perco [1,3] and Matthias Ley [1,2],*

[1] Delta4 GmbH, Vienna, Austria
[2] Division of Pediatric Nephrology and Gastroenterology, Department of Pediatrics and Adolescent Medicine, Comprehensive Center for Pediatrics, Medical University Vienna, Vienna, Austria
[3] Department of Internal Medicine IV, Medical University Innsbruck, Innsbruck, Austria
* Correspondence: matthias.ley@delta4.ai
† These authors contributed equally to this work.

Abstract

The rapid expansion of biomedical literature demands automated summarization tools that can reliably condense research articles into concise, accurate overviews. We benchmarked 62 summarization systems – ranging from frequency-based and TextRank extractors to modern Encoder-Decoder Models (EDMs) and Large Language Models (LLMs) – on a curated set of 1,000 biomedical abstracts paired with author-generated highlights sections as reference summaries. Models were evaluated using a composite suite of metrics covering lexical overlap (ROUGE-1/2/L, BLEU, METEOR), embedding-based semantic similarity (RoBERTa, DeBERTa, all-mpnet-base-v2), and factual consistency (AlignScore). Our results indicate that general-purpose Language Models (LMs) achieve the highest overall scores across both lexical and semantic metrics, outperforming both reasoning-oriented and domain-specific models. Within the general-purpose group, medium-sized models, typically runnable on a single node, often outperform frontier-scale counterparts, suggesting an optimal balance between model capacity and computational efficiency. Statistical extractive methods lag behind all neural approaches. These findings provide a systematic reference for selecting summarization tools in biomedical research and highlight that broad pretraining remains more effective than narrow domain adaptation for generating high-quality scientific summaries.

Keywords: benchmarking; natural language processing; scientific text summarization; large language models; biomedical literature

Received:
Revised:
Accepted:
Published:

Citation: . Title. *Int. J. Mol. Sci.* **2025**, *1*, 0. https://doi.org/

1. Introduction

The exponential growth of scientific literature has created a demand for text summarization methods to support scientists in efficiently prioritizing papers, extracting relevant information, and interpreting complex findings. Automatic Text Summarization (ATS) methods have evolved from statistical approaches to deep learning-based models, becoming increasingly sophisticated and reliable at capturing essential parts from complex research articles. ATS methods have been previously evaluated and described [1,2], but few are tailored for scientific literature summarization [3,4].

1.1. Statistical and Encoder-Decoder Approaches

The pre-neural era of text summarization was mainly characterized by extractive approaches, where in an unsupervised way, summaries were generated by using word or concept frequencies to identify relevant sentences. The first word-frequency based approaches were discussed by Luhn [5], who presented a method based on the assumption that recurrent words in a text are likely more important. Later, Edmundson [6] introduced concepts such as cue words, title words, and sentence position to further enhance the automatic summarization process. The concept of Term Frequency-Inverse Document Frequency (TF-IDF) was later adopted [7] and applied to text summarization by representing sentences as term-weight vectors that down-weight frequently occurring terms with low context specificity in an entire corpus of documents, while promoting rarer terms that at the same time are more context-specific. Word-frequency based approaches have been extensively used in scientific text summarization, being at the basis of more sophisticated strategies [8]. Finally, graph-based statistical methods, in which sentences are represented as nodes and their relationships as edges weighted by similarity measures (e.g. cosine similarity of TF-IDF vectors), allow for the identification of the most central information based on the documents global structure rather than local word counts. A widely used approach in the biomedical domain is TextRank, which constructs a sentence graph to then apply the PageRank algorithm to compute importance sentence importance scores, and generates the summary by selecting the top-ranked sentences [9].

With the advent of Sequence-to-Sequence (Seq2seq) frameworks, summarization shifted toward neural approaches that paraphrase and condense text using Encoder-Decoder architectures, originally implemented with Recurrent Neural Networkss (RNNs), Long Short-Term Memory (LSTM) networks, and Gated Recurrent Units (GRUs) [10,11]. The introduction of self-attention mechanisms replaced Seq2seq frameworks by processing sequences in parallel rather than sequentially, enabling the capture of complex linguistic patterns and long-range contextual relationships [12]. This innovation laid the foundation for transformer architectures that quickly gained popularity in performing a wide range of Natural Language Processing (NLP) tasks, including text summarization. One of the earliest and most influential transformer-based models, Bidirectional Encoder Representations from Transformers (BERT) [13], was widely adopted in domain specific tasks owing to its possibility to be fine-tuned by adding a task-specific output layer. Inspired by BERT's architecture, several abstractive summarization models emerged, including Bidirectional and Auto-Regressive Transformer (BART) - a denoising autoencoder for pretraining Seq2seq models [14] that can be trained or fine-tuned on scientific literature [15,16]. The Text-to-Text Transfer Transformer (T5) model was introduced as a unified text-to-text framework for a broad spectrum of NLP tasks due to its high flexibility with no need for architectural changes [17]. Pre-training with Extracted Gap-sentences for Abstractive Summarization Sequence-to-sequence (PEGASUS), was specifically proposed for abstractive summarization [18] and has been adapted for scientific text with domain-specific variants including "google/pegasus-pubmed" and "google/bigbird-pegasus-large-pubmed". Robustly Optimized BERT Approach (RoBERTa) is an optimized version of BERT trained on a bigger corpus of text, which led to the creation of Longformer [19], a transformer-based architecture that can handle longer texts for text-to-text generation, with "allenai/led-base-16384" and "led-large-16384-arxiv" as notable examples [20].

1.2. Language Models

Despite these advances, the field of ATS quickly moved towards decoder-only architectures which are at the basis of LLMs, able to capture semantic relations with higher flexibility and specificity. LLMs can be classified as (i) general-purpose models, which leverage their broad domain knowledge across diverse NLP tasks, (ii) reasoning-oriented models, characterized by logical text understanding through iterative chain-of-thought processing and instruction tuning [21], and (iii) domain-specific models, tailored for specialized tasks or scientific domains. Several families of LLMs have been developed, including the Generative Pre-trained Transformer (GPT) series developed by OpenAI (GPT-1 [22] through GPT-5) and open-source variants like GPT:OSS, all pre-trained on large-scale text corpora through self-supervised learning. Similarly, Anthropic's Claude Models are built on transformer architecture and trained through a Constitutional AI approach [23]. This family also includes a series of reasoning models such as Sonnet-4 and Opus-4. Meta's Large Language Model Meta AI (Llama) family [24], with LLaMA 3.1 as the most capable open-source model available to date, includes domain-specific adaptations such as OpenBioLLM-LLaMA-3 [25], a biomedical variant trained on a large corpus of high-quality biomedical data, and MedLLaMA-2 [26], a medical LM based on LLaMA 2 architecture. Google developed a series of lightweight models including the Gemma series [27], with Gemma3 as its latest and most powerful reasoning model. Microsoft introduced the Phi series [28,29], which comprises Phi-4-reasoning and Phi-4-mini-reasoning, alongside BioGPT [30], a domain-specific model built on the GPT architecture and fine-tuned for biomedical applications. IBM released the Granite series [31], with Granite 4.0 as its reasoning-capable variant. Mistral AI developed the Mistral family [32], including Magistral as its first reasoning model [33], and Biomistral [34], an open-source variant pretrained on PubMed Central data for biomedical text processing. Alibaba Cloud introduced the Qwen 3 series [35] as an open-source LLM family, which inspired SciLitLLM [36], a specialized model for scientific literature understanding based on Qwen2.5 and trained through Continual Pre-Training (CPT) and Supervised Fine-Tuning (SFT) on scientific literature [36]. DeepSeek has developed Reinforcement Learning (RL)-driven reasoning models that achieve performance comparable to state-of-the-art closed-source models while requiring only a fraction of their training costs [37]. Lastly, Apertus [38] represents Switzerland's first large-scale open, multilingual LM with a fully documented and openly accessible development process.

To the best of our knowledge, no comprehensive benchmarking of text summarization models on biomedical literature has been reported to date. This study addresses this gap by systematically evaluating 62 summarization models, ranging from word-frequency methods to state-of-the-art LLMs, using a curated dataset of 1,000 biomedical abstracts and corresponding highlights sections as reference summaries for benchmarking. By identifying the strengths and limitations of each approach, we provide actionable insights for selecting appropriate summarization tools to accelerate knowledge discovery in biomedical sciences.

2. Materials and Methods

2.1. Becnhmarking Workflow Overview

Figure 1 provides an overview of the complete benchmarking workflow employed in this study, from dataset construction and model inference to metric computation, score aggregation, and statistical analysis.

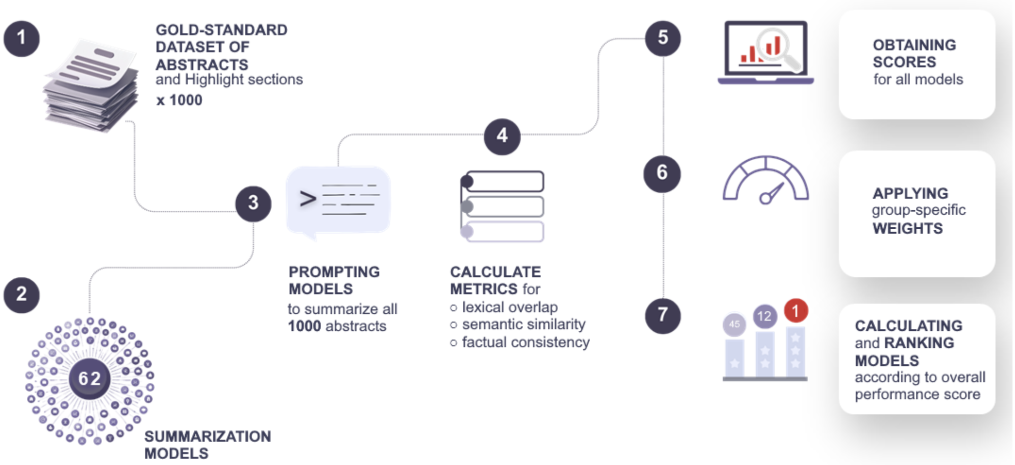


Figure 1. Overview of the benchmarking workflow. (1) Construction of a gold-standard dataset comprising 1,000 biomedical abstracts paired with author-provided highlights sections. (2) Suite of 62 diverse summarization models. (3) Uniform prompting of all models to generate summaries for each abstract. (4–5) Computation of summary quality metrics capturing lexical overlap, semantic similarity, and factual consistency. (6) Aggregation of metrics using group-specific weights to obtain an overall performance score. (7) Ranking of models based on the resulting overall performance scores.

2.2. Gold-Standard Dataset

We generated a gold-standard benchmarking dataset comprising 1,000 biomedical peer-reviewed articles from *ScienceDirect* and *Cell Press* - as these publishers provide a standardized highlights section for each publication [39,40]. This section provides concise bullet points capturing the main findings of each article. The concatenated highlights served as reference summaries in our evaluation, while the corresponding abstracts were used as input texts for the summarization task. While summarization is inherently subjective, author-generated highlights represent the most credible and standardized source, ensuring the captured content reflects the study's intended key messages.

Articles were collected systematically across a variety of journals to ensure coverage of different fields within molecular sciences, including drug discovery, genomics, proteomics, biotechnology, and biochemistry. We selected 50 articles from each of 20 different journals across the two publishers, resulting in 1,000 papers as shown in Table 1.

Table 1. Overview of journals included in the gold-standard dataset. Each journal contributed 50 articles, resulting in 500 articles from *ScienceDirect* and 500 articles from *Cell Press*.

Publisher	Journals
ScienceDirect	Drug Discovery Today; Journal of Molecular Biology; FEBS Letters; Journal of Biotechnology; Gene; Genomics; Journal of Proteomics; The International Journal of Biochemistry & Cell Biology; Cytokine; Developmental Cell
Cell Press	Cell; Cancer Cell; Cell Chemical Biology; Cell Genomics; Cell Host & Microbe; Cell Metabolism; Cell Reports; Cell Reports Medicine; Cell Stem Cell; Cell Systems

This setup provides standardized pairs of abstracts and reference summaries that can be directly used for evaluating automatic summarization methods.

2.3. Summarization Methods

We evaluated 62 summarization models, ranging from simple frequency-based algorithms to Small Language Models (SLMs) & LLMs. These models were classified into five categories, as listed in Table 2. We obtained the pre-trained EDMs through the Hugging Face library, selecting architectures widely used for abstractive summarization tasks to represent well established neural approaches within out benchmark. Additionally, we evaluated a range of popular SLMs and LLMs, defining SLMs as models with fewer than 10 billion parameters [41]. Both SLMs and LLMs with advanced reasoning capabilites were categorized as "reasoning-oriented" to investigate how multi-step problem solving affects summarization performance. Similarly, models fine-tuned on scientific/biomedical data or tailored specifically for text summarization were classified as "domain-specific" to assess the impact of domain adaptation. Overall, this selection covers various model sizes and release dates, ensuring a representative mix of both widely adopted and recent architectures.

Exceptionally large models, such as LLaMA 3.1 405B, were excluded as their computational requirements exceed those of practical summarization pipelines. Each of the 62 models was tasked with generating summaries for the 1,000 abstracts in the dataset, resulting in a total of 62,000 generated summaries for evaluation.

Table 2. Overview of summarization methods/models evaluated in this study, organized by category.

Category	**Methods/Models**
Traditional models	textrank; frequency
General-purpose EDMs	facebook/bart-base; google-t5/t5-base; google-t5/t5-large; google/pegasus-large
Domain-specific EDMs	facebook/bart-large-cnn; google/pegasusxsum; google/pegasus-cnn_dailymail; google/pegasus-pubmed; google/bigbird-pegasuslarge-pubmed; csebuetnlp/mT5_multilingual_XLSum; led_large_16384_arxiv_summarization
General-purpose SLMs	gemma3:270M; gemma3:1b; gemma3:4b; PetrosStav/gemma3-tools:4b; granite3.3:2b; granite3.3:8b; granite4:tiny-h; granite4:small-h; granite4:micro; granite4:micro-h; llama3.1:8b; llama3.2:1b; llama3.2:3b; mistral:7b; phi3:3.8b; gpt-4o-mini; gpt-4.1-mini; chat_swiss-ai/Apertus-8B-Instruct-2509
General-purpose LLMs	gemma3:12b; mistral-nemo:12b; mistral-small3.2:24b; mistral-small-2506; mistral-medium-2505; mistral-large-2411; phi4:14b; gpt-3.5-turbo; gpt-4o; gpt-4.1; claude-3-5-haiku-20241022
Reasoning-oriented SLMs	deepseek-r1:1.5b; deepseek-r1:7b; deepseek-r1:8b; qwen3:4b; qwen3:8b;
Reasoning-oriented LLMs	deepseek-r1:14b; gpt-oss:20b; gpt-5-nano-2025-08-07; gpt-5-mini-2025-08-07; gpt-5-2025-08-07; claude-sonnet-4-20250514; claude-opus-4-20250514; claude-opus-4-1-20250805; magistral-medium-2509
Domain-specific SLMs	completion_microsoft/biogpt; medllama2:7b; chat_aaditya/OpenBioLLM-Llama3-8B; conversational_BioMistral/BioMistral-7B; chat_Uni-SMART/SciLitLLM1.5-7B
Domain-specific LLMs	chat_Uni-SMART/SciLitLLM1.5-14B

2.4. Prompt Design

To ensure comparability across models, we prompted all summarization systems with an identical task description. The prompt instructed the models to generate concise summaries focused on the main findings of each publication while excluding unnecessary background or methodological details. Each model received the publication title and abstract as an input and was asked to produce an output of 15–100 words. If the abstract did not contain any substantive results or conclusions, the model was instructed to return the predefined token INSUFFICIENT_FINDINGS.

The exact prompt used for all models was as follows:

```
Summarize the provided publication (title and abstract) in 15-100 words.

Key requirements:
- Identify main findings, results, or contributions
- Preserve essential context and nuance
- Exclude background, methods unless crucial to conclusions
- Write concisely and objectively
- Avoid repetition and unnecessary qualifiers

```

```
If no substantial findings exist, respond: 'INSUFFICIENT_FINDINGS'
```

2.5. Evaluation Metrics

As there is no single metric that can fully reflect summary quality, especially in the biomedical field where both coverage of key information and factual correctness are critical, we employed both surface-level metrics based on lexical overlap, referred to as lexical-based metrics, and embedding-level metrics. The latter include metrics based on semantic similarity, denoted as semantic-based metrics, as well as one metric that evaluates factual consistency. Since 1,000 abstracts were used in this study, each metric generates a vector of 1,000 values, from which an average value was calculated. These values were then used to build a score that reflects the true summary quality.

2.5.1. Surface-level Metrics

Surface-level metrics compare the generated summaries with the reference summaries mainly at word or phrase level. While they do not capture meaning beyond surface overlap, they remain common metrics in summarization research and provide a straightforward foundation for evaluation. We used three Recall-Oriented Understudy for Gisting Evaluation (ROUGE) variants (ROUGE-1, ROUGE-2, ROUGE-L) [42], Bilingual Evaluation Understudy (BLEU) [43], and Metric for Evaluation of Translation with Explicit ORdering (METEOR) [44]. ROUGE-1 and ROUGE-2 measure how many unigrams (single words) or bigrams (word pairs) from the reference appear in the generated output, while ROUGE-L identifies the longest sequence of words shared between the two. BLEU calculates how many n-grams in the output also occur in the reference, emphasizing precision over recall and applying a brevity penalty to counteract the tendency toward overly short summaries. METEOR extends n-gram matching by considering word stems and synonyms, making it more robust to wording variations. Together, these metrics offer a simple but transparent point of reference.

2.5.2. Embedding-based Metrics

To capture similarity beyond surface-level word overlap, we included a set of embedding-based metrics built on pre-trained transformer models. These methods generate vector representations of text, allowing them to capture semantic similarity rather than just word overlap. We employed RoBERTa [45] and Decoding-enhanced BERT with Disentangled Attention (DeBERTa) [46], two transformer-based models with strong performance across NLP tasks. In summarization evaluation, they can assess whether two summaries capture the same content even if phrased differently.

We further included all-mpnet-base-v2 [47], a transformer model fine-tuned for sentence similarity. Unlike RoBERTa and DeBERTa, which are general-purpose encoders, Masked and Permuted Pre-training (MPNet) was trained with a focus on alignment at the sentence-level. This characteristic makes it a useful complement to the other metrics, as it is particularly sensitive to whether the overall meaning of a reference summary is preserved in the system output.

Finally, to evaluate factual consistency, we applied AlignScore [48], a metric designed to assess whether the statements in a generated summary are supported by the source text. In contrast to the other metrics, AlignScore compares the output to the input text itself (i.e. the publication abstract) rather than the reference summary (i.e. the highlights section), as factual accuracy can only be assessed relative to the original input text. This addition ensures that our evaluation captures errors and hallucinations that might otherwise be overlooked.

2.5.3. Overall Performance Metric

To comprehensively assess the performance of each model on the summarization task, we employed a multi-metric framework covering three dimensions: lexical (n=5), semantic (n=3), and factual (n=1). To prevent the impact of dimension imbalance, we first computed an average score for each category as follows:

$$\text{avg}_{\text{lexical}} = \frac{\text{ROUGE-1} + \text{ROUGE-2} + \text{ROUGE-L} + \text{METEOR} + \text{BLEU}}{5} \tag{1}$$

$$\text{avg}_{\text{semantic}} = \frac{\text{RoBERTa} + \text{DeBERTa} + \text{all-mpnet-base-v2}}{3} \tag{2}$$

$$\text{avg}_{\text{factual}} = \text{AlignScore} \tag{3}$$

In addition, averaging metrics within each dimension ensures that highly correlated measures such as ROUGE-1, ROUGE-2, and ROUGE-L do not dominate the evaluation by hindering the contribution of less correlated metrics that capture related but distinct aspects of summarization (Figure 2). We then used these averaged scores to construct an overall performance score, weighting each dimension according to literature-based evidences. Specifically, research indicates that semantic metrics correlate more strongly with human judgment than lexical ones [49]. Furthermore, we accounted for the fact that AlignScore, due to its different point of reference, tends to favor extractive approaches [50], while showing only a moderate correlation with both lexical and semantic metrics (Figure 2). Based on these considerations, we computed a weighted overall performance score for each model as follows:

$$\text{Overall performance score} = 0.35 \times \text{avg}_{\text{lexical}} + 0.40 \times \text{avg}_{\text{semantic}} + 0.25 \times \text{avg}_{\text{factual}} \tag{4}$$

Finally, we ranked the models based on their overall performance scores, constructing the performance rank, with lower ranks indicating better performance. To examine which model performed best for each dimension (lexical, semantic, factual), we also ranked each respective average score.

2.6. Statistical Analysis

To assess differences in overall performance between model categories and families, we first applied Welch's Analysis of Variance (ANOVA) and conducted pairwise post-hoc comparisons using the Games-Howell test. This procedure is well suited for groups with unequal sample sizes and heterogeneous variances, conditions that apply to our benchmark due to the heterogeneity of models within each category and family. Adjusted p-values were used to determine the statistical significance of between-group differences, as reported in the Results section.

2.7. Benchmarking Framework

The benchmark was conducted using Python 3.12. Gold standard data were retrieved from open-access publications published by *ScienceDirect* and *Cell Press* through manual extraction of titles, abstracts, and highlight sections, along with metadata including publication URLs, identifiers, section types, and article types where available. All data were stored in machine-readable JSON format.

The framework was implemented using the Python standard library supplemented by several specialized packages: pandas [51] for data import and export, scikit-learn [52] for computing cosine similarities of embeddings and TF-IDF vectors, networkx [53] for graph construction and PageRank algorithm [54]. Additional evaluation metrics were computed using NLTK [55] for METEOR and BLEU scores, ROUGE-score, BERT-score [56], AlignScore, and sentence-transformers [57] with the all-mpnet-base-v2 model.

Communication with proprietary closed-source LLMs was facilitated through the official Python APIs provided by Anthropic, Mistral AI, and OpenAI. Local LLM execution was performed on a workstation equipped with a NVIDIA RTX A4000 GPU (16GB VRAM) running Ollama as a backend service, accessed through its Python API along with the transformers library [58].

All LLMs were configured with a temperature parameter of 0.2 to optimize reproducibility while avoiding completely deterministic outputs. For the latest generation of OpenAI models featuring adaptive reasoning capabilities, the configuration was set to `text.verbosity = low` and `reasoning.effort = minimal`. The full set of parameters and prompts are documented in the `config.py` file in the GitHub repository.

2.8. Data Availability

The complete source code, documentation, gold standard dataset, and processed results are available at:

https://www.github.com/Delta4AI/LLMTextSummarizationBenchmark.

3. Results

Our benchmark results offer a comparative view of summarization performance across all evaluated models on biomedical abstracts. We first examine the correlations among the chosen evaluations metrics and, based on the findings, identify the best-performing model across lexical, semantic, and factual dimensions. Next, we compare model performance across categories and families to identify significant differences. Finally, we present a case study illustrating how concept coverage varies between models.

3.1. Metric Correlations

To examine how the different evaluation metrics relate to each other, we computed pairwise Spearman correlation coefficients across all models (Figure 2).

Strong positive correlations were observed among most lexical-based metrics (ROUGE-1/2/L, METEOR, and BLEU), with the correlation between METEOR and BLEU marking an exception ($\rho = 0.23$). ROUGE variants showed almost identical behavior ($\rho > 0.89$), while BLEU and METEOR demonstrated slightly weaker but still substantial alignment with ROUGE measures ($\rho = 0.53 - 0.79$).

Most semantic-based metrics (RoBERTa, DeBERTa, and all-mpnet-base-v2) showed high internal consistency ($\rho > 0.5$), reflecting their shared focus on semantic similarity. When compared with lexical-based metrics, correlations were moderate to strong in most cases, indicating that both categories capture related but not identical dimensions of summary quality.

In addition, AlignScore, a factual consistency metric, correlated moderately with the semantic-based metrics even though they both are embedding-based ($\rho = 0.35 - 0.5$), as well as with the lexical-based ones ($\rho = 0.2 - 0.41$), which can be attributed to its different point of reference.

Overall, these relationships demonstrate that the various metrics are broadly consistent while providing complementary perspectives. This supports the use of an aggregated “Overall Performance Score” as a balanced indicator of overall summarization performance.

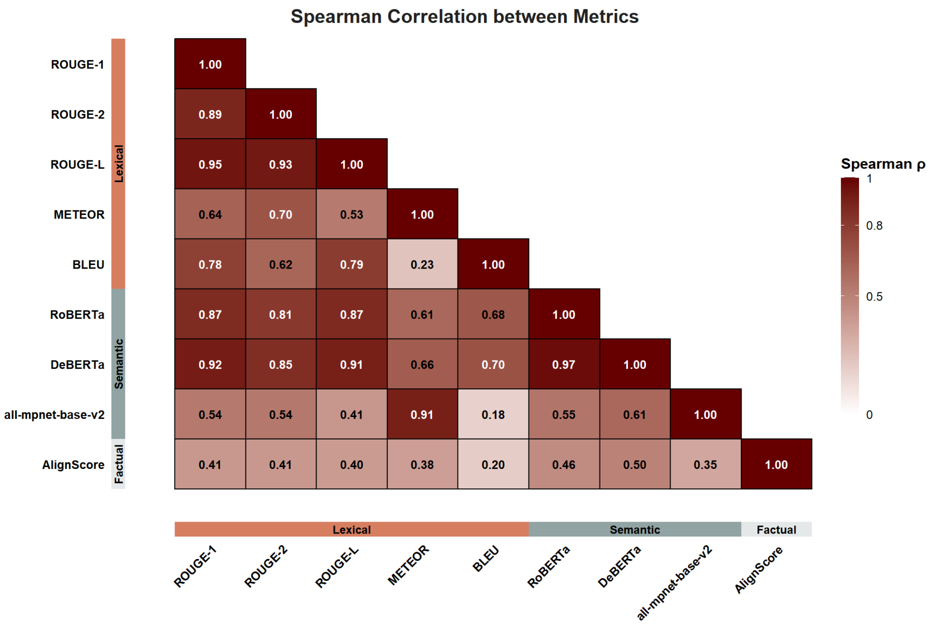


Figure 2. Correlation matrix of evaluation metrics. Each cell in the matrix represents the Spearman correlation coefficient (ρ) between two metrics based on their mean scores across all models. Metric categories (lexical, semantic, factual) are indicated on the x and y axes. For visualization purposes, cells with values higher than 0.8 are shown with white text.

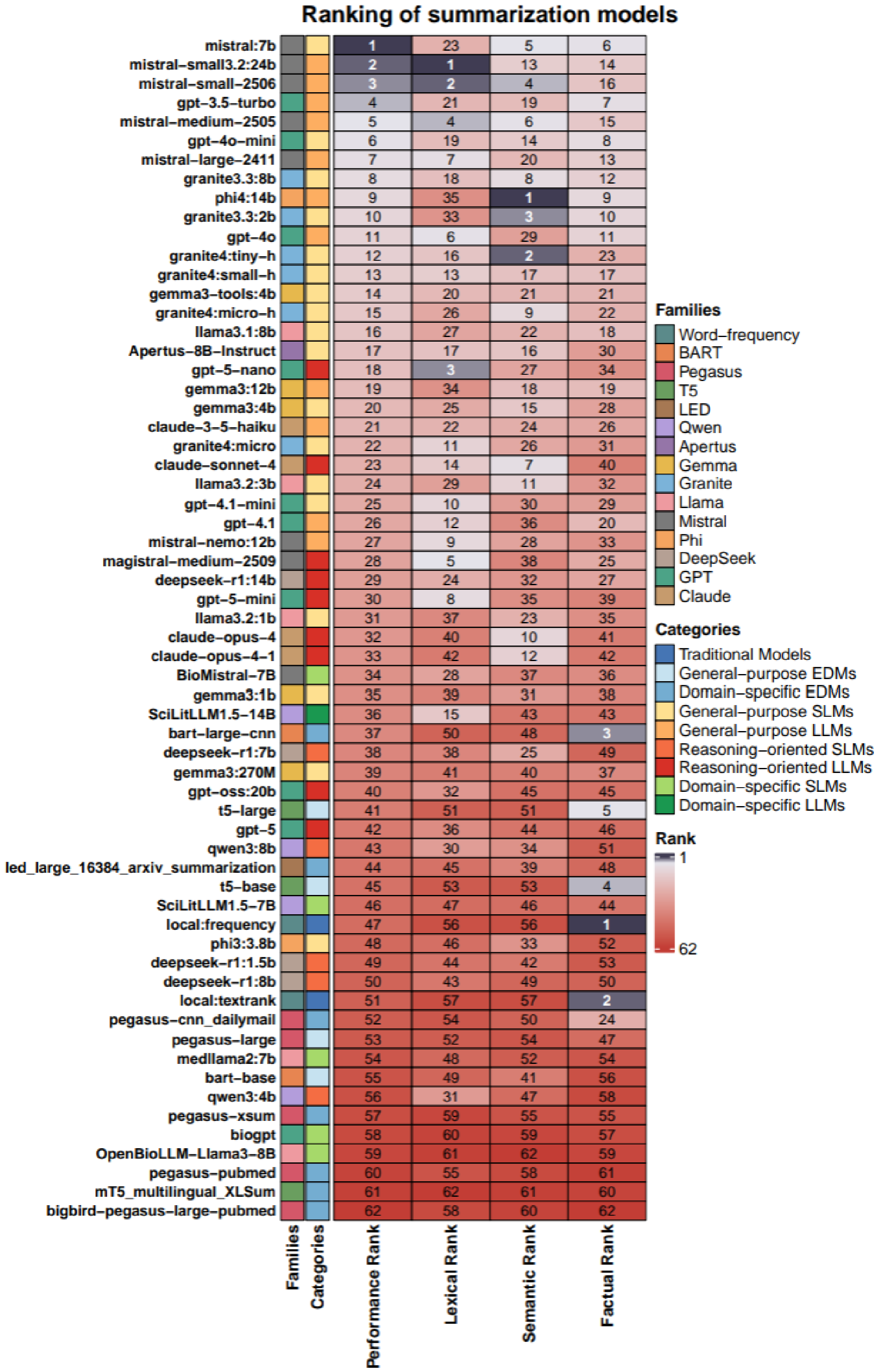
3.2. Overall Model Performance

Based on overall performance ranks, referred to as "Performance Rank" and derived from our multi-metric evaluation framework (2.5.3 Overall Performance Metric), models from the Mistral family were top-ranked with high performance scores across the three evaluated metric dimensions, as depicted in Figure 3.

The mistral:7b model ranked first, followed by mistral-small3.2:24b and mistral-medium-2506. The lowest-ranked models included bigbird-pegasus-large-pubmed and pegasus-pubmed from the PEGASUS family, and mT5_multilingual_XLSum from the T5 family. Overall, domain-specific SLMs and EDM models showed poor performance across all the different dimensions of metrics.

Among the 10 top-ranked models, six were general-purpose LLMs and four were general-purpose SLMs. A similar trend was observed in the top half of the ranking (positions 10 to 32) with the presence of some reasoning-oriented LLMs and with the exception of a single domain-specific LLM ranked at 20 (SciLitLLM1.5-14B). In contrast, in the lower half of the ranking, where models start to perform poorly across most metric dimensions, the majority were reasoning-oriented SLMs, general-purpose EDMs, domain-specific EDMs, and traditional models.

Considering each metric dimension separately, some Mistral models ranked highly in the lexical dimension with GPT-5-nano, a reasoning-oriented LLM from the GPT family, ranked third. In the semantic dimension, Mistral models achieved high ranks, but the highest positions were occupied by the Phi4:14b general-purpose LLM and some models from the Granite family, such as granite4:tiny-h and granite3.3:2b. Finally, the highest ranks based on the factual dimension were covered by traditional and encoder-decoder models. Detailed performance indications for each model across all individual ranked metrics are provided in Figure S1 (supplementary).



	Performance Rank	Lexical Rank	Semantic Rank	Factual Rank
mistral:7b	1	23	5	6
mistral–small3.2:24b	2	1	13	14
mistral–small–2506	3	2	4	16
gpt–3.5–turbo	4	21	19	7
mistral–medium–2505	5	4	6	15
gpt–4o–mini	6	19	14	8
mistral–large–2411	7	7	20	13
granite3.3:8b	8	18	8	12
phi4:14b	9	35	1	9
granite3.3:2b	10	33	3	10
gpt–4o	11	6	29	11
granite4:tiny–h	12	16	2	23
granite4:small–h	13	13	17	17
gemma3–tools:4b	14	20	21	21
granite4:micro–h	15	26	9	22
llama3.1:8b	16	27	22	18
Apertus–8B–Instruct	17	17	16	30
gpt–5–nano	18	3	27	34
gemma3:12b	19	34	18	19
gemma3:4b	20	25	15	28
claude–3–5–haiku	21	22	24	26
granite4:micro	22	11	26	31
claude–sonnet–4	23	14	7	40
llama3.2:3b	24	29	11	32
gpt–4.1–mini	25	10	30	29
gpt–4.1	26	12	36	20
mistral–nemo:12b	27	9	28	33
magistral–medium–2509	28	5	38	25
deepseek–r1:14b	29	24	32	27
gpt–5–mini	30	8	35	39
llama3.2:1b	31	37	23	35
claude–opus–4	32	40	10	41
claude–opus–4–1	33	42	12	42
BioMistral–7B	34	28	37	36
gemma3:1b	35	39	31	38
SciLitLLM1.5–14B	36	15	43	43
bart–large–cnn	37	50	48	3
deepseek–r1:7b	38	38	25	49
gemma3:270M	39	41	40	37
gpt–oss:20b	40	32	45	45
t5–large	41	51	51	5
gpt–5	42	36	44	46
qwen3:8b	43	30	34	51
led_large_16384_arxiv_summarization	44	45	39	48
t5–base	45	53	53	4
SciLitLLM1.5–7B	46	47	46	44
local:frequency	47	56	56	1
phi3:3.8b	48	46	33	52
deepseek–r1:1.5b	49	44	42	53
deepseek–r1:8b	50	43	49	50
local:textrank	51	57	57	2
pegasus–cnn_dailymail	52	54	50	24
pegasus–large	53	52	54	47
medllama2:7b	54	48	52	54
bart–base	55	49	41	56
qwen3:4b	56	31	47	58
pegasus–xsum	57	59	55	55
biogpt	58	60	59	57
OpenBioLLM–Llama3–8B	59	61	62	59
pegasus–pubmed	60	55	58	61
mT5_multilingual_XLSum	61	62	61	60
bigbird–pegasus–large–pubmed	62	58	60	62

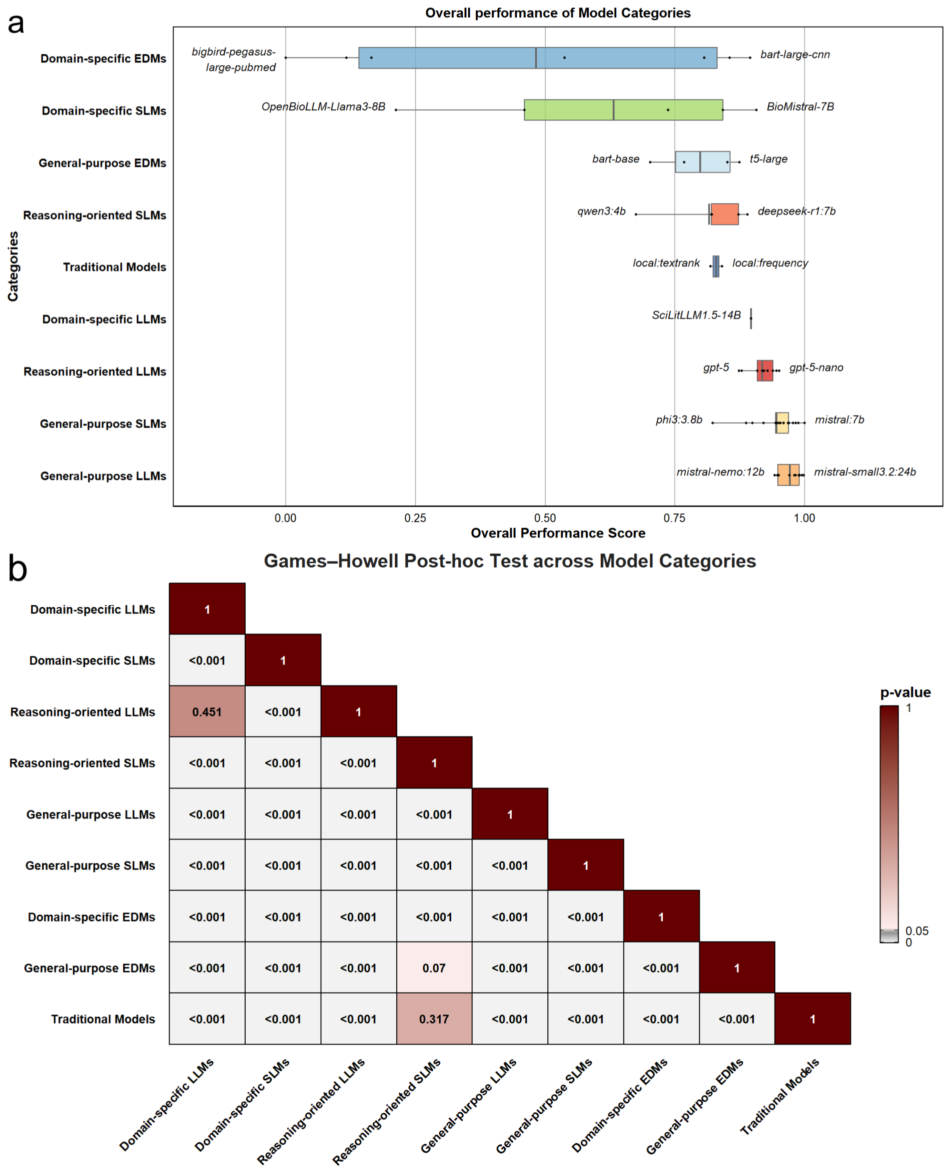
Figure 3. Overview of the performance of all evaluated models across lexical, semantic and factual metric dimensions. Each row corresponds to one model. Columns show the overall Performance Rank and independent ranks for each metric dimension, with lower ranks indicating better performance. The figure displays each model's family, and category. Models are sorted by the Performance Rank. Ranks are displayed using a continuous diverging color gradient, ranging from navy-blue tones for top-performing models through light neutral shades for mid-ranked models to dark red tones for low-performing models. Text color is switched to white for the highest-ranked models (ranks ≤ 3) to improve readability.

3.3. Category Comparisons

To compare model performance across categories, we displayed the distribution of overall performance scores within each category, along with the best- and worst-performing individual models (Figure 4a).

Domain-specific EDMs, domain-specific SLMs, traditional models, and general-purpose EDMs formed the weakest groups, with substantially lower performance scores. Notably, domain specific EDMs and domain-specific SLMs displayed a wide range of results, while all the general-purpose LLMs achieved high overall performance, with general-purpose SLMs being distributed wider but following closely. Reasoning-oriented LLMs also performed competitively, with performance scores comparable to some general-purpose SLMs.

To evaluate whether differences between groups were significant, we conducted a Games-Howell post-hoc comparison as presented in Figure 4b. Its results show that most between-category differences in the overall performance score were statistically significant. Specifically, general-purpose LLMs differed significantly from all lower-performing groups. Only a few comparisons were found not statistically significant: the difference between reasoning-oriented LLMs and domain-specific LLMs, the difference between general-purpose EDMs and reasoning-oriented SLMs, and the difference between traditional models and reasoning-oriented SLMs.



	Domain-specific LLMs	Domain-specific SLMs	Reasoning-oriented LLMs	Reasoning-oriented SLMs	General-purpose LLMs	General-purpose SLMs	Domain-specific EDMs	General-purpose EDMs	Traditional Models
Domain-specific LLMs	1								
Domain-specific SLMs	<0.001	1							
Reasoning-oriented LLMs	0.451	<0.001	1						
Reasoning-oriented SLMs	<0.001	<0.001	<0.001	1					
General-purpose LLMs	<0.001	<0.001	<0.001	<0.001	1				
General-purpose SLMs	<0.001	<0.001	<0.001	<0.001	<0.001	1			
Domain-specific EDMs	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	1		
General-purpose EDMs	<0.001	<0.001	<0.001	0.07	<0.001	<0.001	<0.001	1	
Traditional Models	<0.001	<0.001	<0.001	0.317	<0.001	<0.001	<0.001	<0.001	1

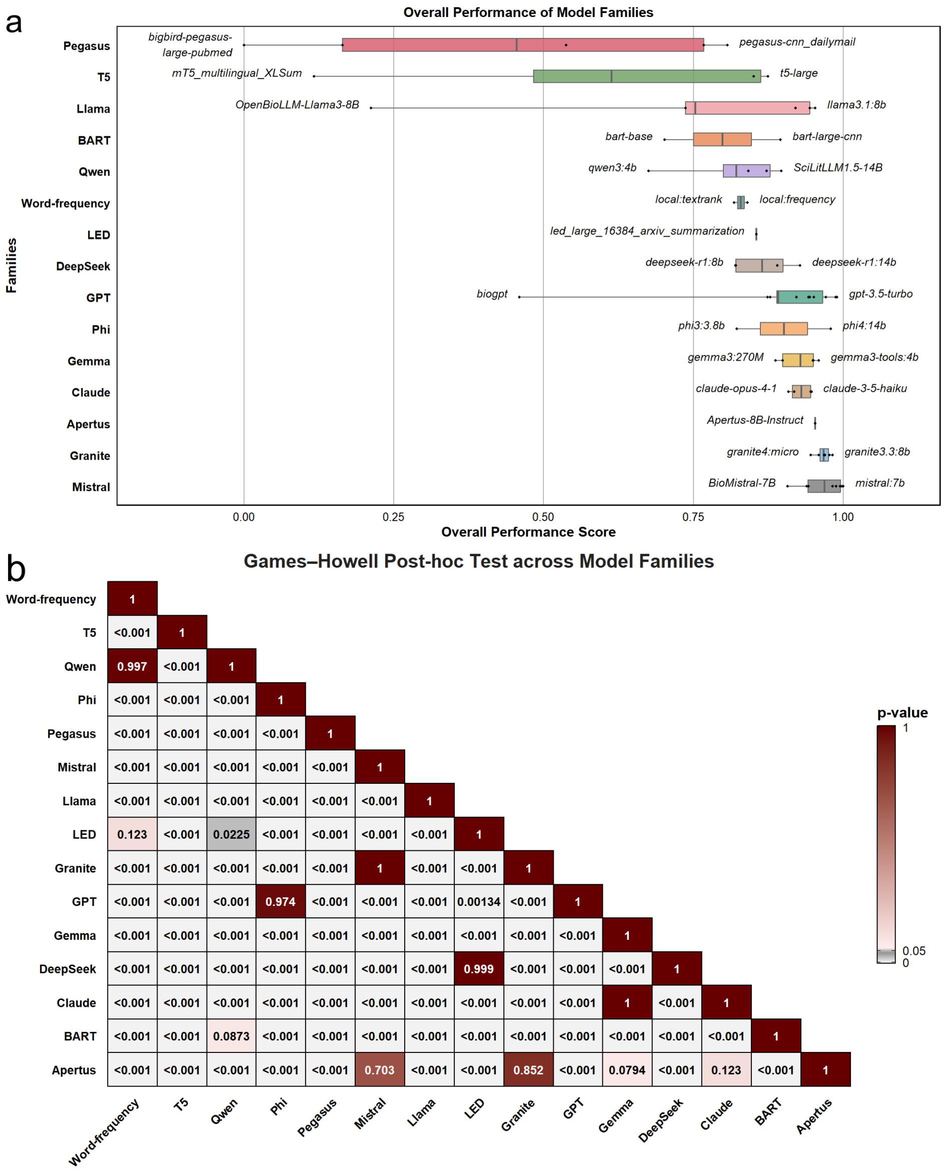
Figure 4. (a) Boxplots summarizing the distribution of overall performance scores across the nine model categories, with each model represented by a black dot. Categories are ordered from lowest- to highest-performing according to the performance score. Moreover, the best- and worst-performing models of each category are highlighted. **(b)** Games–Howell pairwise adjusted p-value matrix comparing all category pairs. Each cell shows the adjusted p-value for the difference in metric mean scores between two categories, with white-to-gray shading indicating smaller p-values.

3.4. Family Comparisons

To complement the category-level findings, we next examined performance across model families, since models within the same family often share architectural features or training strategies that may cause consistent performance trends. Figure 5a illustrates the distribution of overall performance scores across all model families. Similar to the category-level analysis, there were clear performance differences between architectural lineages. The Mistral family achieved the strongest overall results, with several models ranking among the highest-scoring models in the entire benchmark. Families dominated by modern LLM or SLM architectures, such as Granite, Claude, and Gemma, consistently outperformed more traditional extractive approaches and families built on encoder-decoder architectures such as Longformer Encoder-Decoder (LED), BART, T5 and PEGASUS.

In particular, the GPT and Llama families showed lower performance scores than expected given the competitive performance of their top models. This was primarily due to the inclusion of domain-specific small models, such as BioGPT and OpenBioLLM-Llama3-8B, which performed substantially worse than their general-purpose counterparts and therefore pulled down the family-level averages.

Regarding the Games-Howell post-hoc matrix in the family comparison (Figure 5b), a combination of significant and non-significant differences between families was observed, reflecting the greater heterogeneity within some lineages. While many high-performing families differed significantly from traditional and encoder-decoder dominated families, several comparisons among modern SLMs and LLMs dominated families did not reach statistical significance.



	Word-frequency	T5	Qwen	Phi	Pegasus	Mistral	Llama	LED	Granite	GPT	Gemma	DeepSeek	Claude	BART	Apertus
Word-frequency	1														
T5	<0.001	1													
Qwen	0.997	<0.001	1												
Phi	<0.001	<0.001	<0.001	1											
Pegasus	<0.001	<0.001	<0.001	<0.001	1										
Mistral	<0.001	<0.001	<0.001	<0.001	<0.001	1									
Llama	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	1								
LED	0.123	<0.001	0.0225	<0.001	<0.001	<0.001	<0.001	1							
Granite	<0.001	<0.001	<0.001	<0.001	<0.001	1	<0.001	<0.001	1						
GPT	<0.001	<0.001	<0.001	0.974	<0.001	<0.001	<0.001	0.00134	<0.001	1					
Gemma	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	1				
DeepSeek	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	0.999	<0.001	<0.001	<0.001	1			
Claude	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	1	<0.001	1		
BART	<0.001	<0.001	0.0873	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	1	
Apertus	<0.001	<0.001	<0.001	<0.001	<0.001	0.703	<0.001	<0.001	0.852	<0.001	0.0794	<0.001	0.123	<0.001	1

Figure 5. a) Boxplots summarizing the distribution of overall performance scores across all model families, with each model represented by a black dot. Families are ordered from lowest- to highest-performing according to the performance score. Moreover, the best- and worst-performing models of each family are highlighted. **b)** Games–Howell pairwise adjusted *p*-value matrix comparing all family pairs. Each cell shows the adjusted *p*-value for the difference in metric mean scores between two families, with white-to-gray shading indicating smaller *p*-values.

3.5. Qualitative Analysis: Case Study

To illustrate how concept coverage differs between models beyond aggregate metrics, we examine two summaries of a biomedical research article [doi:10.1016/j.cell.2025.05.026] against publisher-provided highlights. The source article, titled *"A percolation phase transition controls complement protein coating of surfaces"*, includes four key highlights: (H1) The complement protein network has a switch-like response when attacking surfaces; (H2) Complement "decides" to coat surfaces if surface protein spacing is below a threshold; (H3) Complement's threshold decision-making arises from a percolation phase transition; (H4) Complexity science shows how complement makes discrete decisions attacking surfaces.

The generated summaries are as follows:

- **mistral-small3.2:24b:** "This study reveals that the complement system's activation on materials exhibits a sharp threshold behavior, controlled by the surface density of complement attachment points. This percolation-like phase transition occurs across scales, from nanoparticles to macroscale materials, and is driven by a minimal subnetwork of complement proteins. The findings explain how complex protein networks 'decide' to interact with and attack materials, initiating inflammation."
- **medllama2:7b:** "The main findings are that there is a sharp threshold for complement activation upon varying a fundamental material parameter and this threshold manifests at scales spanning single nanoparticles to macroscale pathologies. The study shows how a minimal subnetwork of complement proteins can give rise to percolation-type critical transitions in the complement response, which explains why some materials are more likely to trigger an immune response than others."

Table 3 summarizes the concept-level coverage and overall performance for both models. While both models capture the switch-like behavior (H1) and the percolation transition (H3), the lower-performing model fails to explicitly link the threshold to surface density/spacing (H2) and omits the complexity-based decision-making framework (H4). In contrast, **mistral-small3.2:24b** achieves full coverage by explicitly connecting the physical density to the "decisions" made by the complex protein network. The overall performance scores, derived using the weighted metric aggregation introduced in Section 2.5.3, further reflect this difference, with **mistral-small3.2:24b** achieving a higher score compared to **medllama2:7b**.

Table 3. Concept coverage analysis of model-generated summaries. Symbols: ✓= fully covered; ∼ = partially covered; × = not covered. Overall performance scores are derived using the weighted metric aggregation described in Section 2.5.3.

Reference Concept	**mistral-small3.2:24b**	**medllama2:7b**
H1: Switch-like response	✓	✓
H2: Surface density threshold	✓	∼
H3: Percolation phase transition	✓	✓
H4: Discrete decision-making	✓	×
Coverage Score	4.0 / 4.0	2.5 / 4.0
Overall Performance Score	0.659	0.512

4. Discussion

Our benchmarking analysis, which evaluated the summarization capabilities of 62 diverse models on a curated dataset of 1,000 abstracts, revealed clear performance differences between the evaluated summarization approaches. General-purpose LLMs achieved the highest summarization quality across all metric dimensions, followed closely by general-purpose SLMs and reasoning-oriented LLMs. In contrast, domain-specific models, encoder–decoder architectures, and traditional extractive methods ranked lower based on overall performance evaluation. These results highlight the clear progression from extractive and encoder–decoder approaches toward transformer-based models, while also showing that domain-specific fine-tuning alone does not necessarily lead to improved summarization quality.

Comparing models by architecture, size, and domain focus shows that, overall, LLMs perform best, likely due to their high number of parameters which enable them to better understand the complex context typical for biomedical literature. While very small models, like Gemma3:270M, can indeed lack the capacity to handle this complexity, SLMs remain competitive, with some models, such as mistral:7b and gpt-4o-mini, even outperforming certain LLMs across the three metric dimensions. This may be attributed to the fact that smaller datasets are often more curated and of higher quality compared to the large amount of data required to train a big model [59].

Interestingly, medium-sized models (e.g many models in the Mistral family), appear to be more performant than larger proprietary ones. These models seem to reach an optimal compromise on number of parameters and overall performance where additional parameters could disrupt this equilibrium, leading to potentially over-fitting or plateauing performance [60].

Another interesting result of our analysis was that overall general-purpose models outperform both domain-specific models specialized in the biomedical domain and the one specialized for text summarization, regardless of model size. A possible explanation for this behavior is that domain-specific models fine-tuned on biomedical text, might be better for learning and understanding the complex biomedical terminology or lexical patterns but might fail in summarization tasks. On the other hand, models specifically designed for text summarization might be good at summarizing in general but fail at capturing the complex biomedical meaning. That is why generalist models, leveraging their broad knowledge, seem to perform better [61]. Additionally, domain-specific models can “forget” the general knowledge that was acquired during the pre-training phase, experiencing a phenomenon called “catastrophic forgetting”, which represents an issue when the task requires both domain-specific knowledge, biomedical knowledge, and context understanding for text summarization [62].

Most reasoning-oriented models ranked in the middle, indicating moderate performance. This can be explained by the intrinsic multi-step logical reasoning nature of these models, that, while it can be advantageous for tasks that require breaking problems down into sequential steps like for mathematics or coding, it may not be ideal for text summarization that requires semantic compression and factual grounding instead [63].

Even though our results are based on a robust evaluation framework, there are several factors worth discussing. Model access methods varied across the evaluation due to differing Application Programming Interface (API) capabilities and requirements. HuggingFace models were accessed through their supported interfaces: the pipeline API (task="summarization") where available, or chat/completion formats for models that did not support the pipeline approach. Ollama models required use of the generate endpoint with merged prompts, while OpenAI, Anthropic, and Mistral models each mandated their respective provider-specific APIs (responses.create, messages.create, and chat.complete) with distinct message structures. We applied hyperparameter normalization where possible, though API-level constraints prevented full standardization. For example, GPT-5 does not support temperature control, instead offering only reasoning-specific parameters. Additionally, proprietary middleware layers may transform requests and responses in undocumented ways, potentially affecting outputs independently of the underlying model architectures. These necessary methodological variations warrant consideration when interpreting performance differences across models.

Regarding limitations it is also important to note that this benchmark focuses on a single summarization task: generating concise summaries from biomedical abstracts. This setup provides a clear and well-defined evaluation framework, but the findings may not fully extend to other forms of scientific or biomedical summarization, including full-length articles, clinical trial data, or lay-oriented summaries. Given the rapid evolution of LLMs, these results just capture a specific snapshot in time and may change as newer architectures and models become available.

Another limitation lies in the exclusive reliance on automatic evaluation metrics. Although combining lexical-based, semantic-based, and factual consistency measures offers a broad view, human assessment could provide a more nuanced understanding of readability, coherence, and factual correctness. Future work could therefore extend this benchmark by integrating expert-based evaluations, exploring alternative summarization tasks, and including emerging model families as they are released.

The results of this benchmark provide useful guidance for selecting summarization models in biomedical and scientific settings. The strong performance of general-purpose LMs indicates that broad, diverse pretraining is often more advantageous than narrow domain adaptation when dealing with unseen scientific content.

Another key consideration is the trade-off between output quality and processing efficiency. While LLMs achieved the highest overall scores, SLMs deliver competitive results at substantially lower computational cost [64], which makes them especially attractive for large-scale or resource-constrained applications. Choosing between large and small models therefore depends not only on desired output quality but also on the intended scale of summarization.

Overall, the findings suggest that general-purpose LMs currently offer the most reliable and practical choice for biomedical text summarization. Their consistent performance across evaluation criteria demonstrates that broad generalization outweighs the marginal gains from more narrowly specialized or fine-tuned approaches, many of which are not primarily optimized for text summarization.

5. Conclusions

This benchmark provides a comprehensive and up-to-date evaluation of 62 summarization methods on a curated dataset of 1,000 biomedical abstracts paired with standardized highlights sections. Across all lexical overlap, semantic similarity, and factual consistency metrics, general-purpose LLMs demonstrated the most consistent performances. Their ability to incorporate broad pretraining with strong semantic understanding, enabled them to outperform reasoning-oriented models, domain-specific models, and encoder-decoder architectures. Especially medium-sized models achieved the highest overall rankings, which indicates that increased scale beyond this range does not automatically translate to improved summarization quality for biomedical text.

The results also highlight that SLMs remain competitive for settings where computational resources are limited as they offer balance between efficiency and output quality. In contrast, domain-specific models did not show a systematic advantage, showing that narrow fine-tuning alone is insufficient to match the generalization of broadly trained LLMs for this task.

Overall, this benchmark provides a systematic reference point for selecting summarization models in biomedical research. By assessing a broad spectrum of architectures under a unified framework we underline the strengths and limitations of currently available systems and highlight the superiority of general-purpose LLMs for scientific text summarization. These insights can support researchers and practitioners in choosing models that balance output quality, computational cost, and practical usability in biomedical workflows.

Abbreviations

The following abbreviations are used in this manuscript:

LM	Language Model
LLM	Large Language Model
SLM	Small Language Model
EDM	Encoder-Decoder Model
NLP	Natural Language Processing
ATS	Automatic Text Summarization
TF-IDF	Term Frequency-Inverse Document Frequency
Seq2seq	Sequence-to-Sequence
RNN	Recurrent Neural Networks
LSTM	Long Short-Term Memory
GRU	Gated Recurrent Unit
CPT	Continual Pre-Training
SFT	Supervised Fine-Tuning
RL	Reinforcement Learning
API	Application Programming Interface
ANOVA	Analysis of Variance
BERT	Bidirectional Encoder Representations from Transformers
BART	Bidirectional and Auto-Regressive Transformer
T5	Text-to-Text Transfer Transformer
PEGASUS	Pre-training with Extracted Gap-sentences for Abstractive Summarization Sequence-to-sequence
GPT	Generative Pre-trained Transformer
Llama	Large Language Model Meta AI
LED	Longformer Encoder-Decoder
ROUGE	Recall-Oriented Understudy for Gisting Evaluation
BLEU	Bilingual Evaluation Understudy
METEOR	Metric for Evaluation of Translation with Explicit ORdering
RoBERTa	Robustly Optimized BERT Approach
DeBERTa	Decoding-enhanced BERT with Disentangled Attention
MPNet	Masked and Permuted Pre-training

References

1. Zhang, Y.; Jin, H.; Meng, D.; Wang, J.; Tan, J. A Comprehensive Survey on Process-Oriented Automatic Text Summarization with Exploration of LLM-Based Methods, 2025, [arXiv:cs.AI/2403.02901].
2. Zhang, H.; Yu, P.S.; Zhang, J. A Systematic Survey of Text Summarization: From Statistical Methods to Large Language Models, 2024, [arXiv:cs.CL/2406.11289].
3. Rohil, M.K.; Magotra, V. An exploratory study of automatic text summarization in biomedical and healthcare domain. *Healthcare Analytics* **2022**, *2*, 100058. https://doi.org/https://doi.org/10.1016/j.health.2022.100058.

4. Xie, Q.; Luo, Z.; Wang, B.; Ananiadou, S. A Survey for Biomedical Text Summarization: From Pre-trained to Large Language Models, 2023, [arXiv:cs.CL/2304.08763].
5. Luhn, H.P. The Automatic Creation of Literature Abstracts. *IBM J. Res. Dev.* **1958**, *2*, 159–165.
6. Edmundson, H.P. New Methods in Automatic Extracting. *J. ACM* **1969**, *16*, 264–285. https://doi.org/10.1145/321510.321519.
7. Robertson, S. Understanding Inverse Document Frequency: On Theoretical Arguments for IDF. *Journal of Documentation - J DOC* **2004**, *60*, 503–520. https://doi.org/10.1108/00220410410560582.
8. Reeve, L.H.; Han, H.; Nagori, S.V.; Yang, J.C.; Schwimmer, T.A.; Brooks, A.D. Concept frequency distribution in biomedical text summarization. In Proceedings of the Proceedings of the 15th ACM International Conference on Information and Knowledge Management, New York, NY, USA, 2006; CIKM '06, p. 604–611. https://doi.org/10.1145/1183614.1183701.
9. Mihalcea, R.; Tarau, P. TextRank: Bringing Order into Text. In Proceedings of the Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing; Lin, D.; Wu, D., Eds., Barcelona, Spain, 2004; pp. 404–411.
10. Afzal, M.; Alam, F.; Malik, K.M.; Malik, G.M. Clinical Context–Aware Biomedical Text Summarization Using Deep Neural Network: Model Development and Validation. *J Med Internet Res* **2020**, *22*, e19810. https://doi.org/10.2196/19810.
11. Almasoud, A.; Hassine, S.; Al-Wesabi, F.; Nour, M.; Hilal, A.; Al Duhayyim, M.; Hamza, A.; Motwakel, A. Automated Multi-Document Biomedical Text Summarization Using Deep Learning Model. *Computers, Materials & Continua* **2022**, *71*, 5800. https://doi.org/10.32604/cmc.2022.024556.
12. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need, 2023, [arXiv:cs.CL/1706.03762].
13. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, 2019, [arXiv:cs.CL/1810.04805].
14. Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; Zettlemoyer, L. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. *CoRR* **2019**, *abs/1910.13461*, [1910.13461].
15. Yuan, H.; Yuan, Z.; Gan, R.; Zhang, J.; Xie, Y.; Yu, S. BioBART: Pretraining and Evaluation of A Biomedical Generative Language Model. In Proceedings of the Proceedings of the 21st Workshop on Biomedical Language Processing; Demner-Fushman, D.; Cohen, K.B.; Ananiadou, S.; Tsujii, J., Eds., Dublin, Ireland, 2022; pp. 97–109. https://doi.org/10.18653/v1/2022.bionlp-1.9.
16. Abinaya, S.; Vigil, M.; Keerthika, K.; Varshasri, R. Medical Text Summarization Using BART with LoRA-Based Parameter Efficient Fine Tuning **2024**.
17. Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; Liu, P.J. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research* **2020**, *21*, 1–67.
18. Zhang, J.; Zhao, Y.; Saleh, M.; Liu, P.J. PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization, 2020, [arXiv:cs.CL/1912.08777].
19. Beltagy, I.; Peters, M.E.; Cohan, A. Longformer: The Long-Document Transformer. *arXiv:2004.05150* **2020**.
20. Steblianko, O.; Shymkovych, V.; Kravets, P.; Novatskyi, A.; Shymkovych, L. Scientific article summarization model with unbounded input length. *Information, Computing and Intelligent systems* **2024**, pp. 150–158. https://doi.org/10.20535/2786-8729.5.2024.314724.
21. Plaat, A.; Wong, A.; Verberne, S.; Broekens, J.; van Stein, N.; Back, T. Multi-Step Reasoning with Large Language Models, a Survey, 2025, [arXiv:cs.AI/2407.11511].
22. Radford, A.; Narasimhan, K. Improving Language Understanding by Generative Pre-Training. 2018.
23. Bai, Y.; Kadavath, S.; Kundu, S.; Askell, A.; Kernion, J.; Jones, A.; Chen, A.; Goldie, A.; Mirhoseini, A.; McKinnon, C.; et al. Constitutional AI: Harmlessness from AI Feedback, 2022, [arXiv:cs.CL/2212.08073].
24. Grattafiori, A.; Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Vaughan, A.; et al. The Llama 3 Herd of Models, 2024, [arXiv:cs.AI/2407.21783].
25. Ankit Pal, M.S. OpenBioLLMs: Advancing Open-Source Large Language Models for Healthcare and Life Sciences. https://huggingface.co/aaditya/OpenBioLLM-Llama3-70B, 2024.
26. Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. Llama 2: Open Foundation and Fine-Tuned Chat Models, 2023, [arXiv:cs.CL/2307.09288].
27. Team, G.; Kamath, A.; Ferret, J.; Pathak, S.; Vieillard, N.; Merhej, R.; Perrin, S.; Matejovicova, T.; Ramé, A.; Rivière, M.; et al. Gemma 3 Technical Report, 2025, [arXiv:cs.CL/2503.19786].
28. Abdin, M.; Aneja, J.; Awadalla, H.; Awadallah, A.; Awan, A.A.; Bach, N.; Bahree, A.; Bakhtiari, A.; Bao, J.; Behl, H.; et al. Phi-3 Technical Report: A Highly Capable Language Model Locally on Your Phone, 2024, [arXiv:cs.CL/2404.14219].
29. Abdin, M.; Aneja, J.; Behl, H.; Bubeck, S.; Eldan, R.; Gunasekar, S.; Harrison, M.; Hewett, R.J.; Javaheripi, M.; Kauffmann, P.; et al. Phi-4 Technical Report, 2024, [arXiv:cs.CL/2412.08905].

30. Luo, R.; Sun, L.; Xia, Y.; Qin, T.; Zhang, S.; Poon, H.; Liu, T.Y. BioGPT: generative pre-trained transformer for biomedical text generation and mining. *Briefings in Bioinformatics* **2022**, *23*, [https://academic.oup.com/bib/article-pdf/23/6/bbac409/47144271/bbac409.pdf]. bbac409, https://doi.org/10.1093/bib/bbac409.
31. Mishra, M.; Stallone, M.; Zhang, G.; Shen, Y.; Prasad, A.; Soria, A.M.; Merler, M.; Selvam, P.; Surendran, S.; Singh, S.; et al. Granite Code Models: A Family of Open Foundation Models for Code Intelligence, 2024, [arXiv:cs.AI/2405.04324].
32. Jiang, A.Q.; Sablayrolles, A.; Mensch, A.; Bamford, C.; Chaplot, D.S.; de las Casas, D.; Bressand, F.; Lengyel, G.; Lample, G.; Saulnier, L.; et al. Mistral 7B, 2023, [arXiv:cs.CL/2310.06825].
33. Mistral-AI.; :.; Rastogi, A.; Jiang, A.Q.; Lo, A.; Berrada, G.; Lample, G.; Rute, J.; Barmentlo, J.; Yadav, K.; et al. Magistral, 2025, [arXiv:cs.CL/2506.10910].
34. Labrak, Y.; Bazoge, A.; Morin, E.; Gourraud, P.A.; Rouvier, M.; Dufour, R. BioMistral: A Collection of Open-Source Pretrained Large Language Models for Medical Domains, 2024, [arXiv:cs.CL/2402.10373].
35. Team, Q. Qwen3 Technical Report, 2025, [arXiv:cs.CL/2505.09388].
36. Li, S.; Huang, J.; Zhuang, J.; Shi, Y.; Cai, X.; Xu, M.; Wang, X.; Zhang, L.; Ke, G.; Cai, H. SciLitLLM: How to Adapt LLMs for Scientific Literature Understanding, 2025, [arXiv:cs.LG/2408.15545].
37. Wang, C.; Kantarcioglu, M. A Review of DeepSeek Models' Key Innovative Techniques, 2025, [arXiv:cs.LG/2503.11486].
38. Hernández-Cano, A.; Hägele, A.; Huang, A.H.; Romanou, A.; Solergibert, A.J.; Pasztor, B.; Messmer, B.; Garbaya, D.; Ďurech, E.F.; Hakimi, I.; et al. Apertus: Democratizing Open and Compliant LLMs for Global Language Environments. https://arxiv.org/abs/2509.14233, 2025.
39. Elsevier. Highlights, 2024. https://www.elsevier.com/researcher/author/tools-and-resources/highlights (accessed: 2025-08-07).
40. Cell Press. Final Submission: Other Components: Highlights, 2024. https://www.cell.com/cell/information-for-authors/final-submission (accessed: 2025-08-07).
41. Belcak, P.; Heinrich, G.; Diao, S.; Fu, Y.; Dong, X.; Muralidharan, S.; Lin, Y.C.; Molchanov, P. Small Language Models are the Future of Agentic AI, 2025, [arXiv:cs.AI/2506.02153].
42. Lin, C.Y. ROUGE: A Package for Automatic Evaluation of Summaries. In Proceedings of the Text Summarization Branches Out, Barcelona, Spain, 2004; pp. 74–81.
43. Papineni, K.; Roukos, S.; Ward, T.; Zhu, W.J. Bleu: a Method for Automatic Evaluation of Machine Translation. In Proceedings of the Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics; Isabelle, P.; Charniak, E.; Lin, D., Eds., Philadelphia, Pennsylvania, USA, 2002; pp. 311–318. https://doi.org/10.3115/1073083.1073135.
44. Banerjee, S.; Lavie, A. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In Proceedings of the Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization; Goldstein, J.; Lavie, A.; Lin, C.Y.; Voss, C., Eds., Ann Arbor, Michigan, 2005; pp. 65–72.
45. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. RoBERTa: A Robustly Optimized BERT Pretraining Approach, 2019, [arXiv:cs.CL/1907.11692].
46. He, P.; Liu, X.; Gao, J.; Chen, W. DeBERTa: Decoding-enhanced BERT with Disentangled Attention, 2021, [arXiv:cs.CL/2006.03654].
47. Song, K.; Tan, X.; Qin, T.; Lu, J.; Liu, T.Y. MPNet: Masked and Permuted Pre-training for Language Understanding, 2020, [arXiv:cs.CL/2004.09297].
48. Zha, Y.; Yang, Y.; Li, R.; Hu, Z. AlignScore: Evaluating Factual Consistency with a Unified Alignment Function, 2023, [arXiv:cs.CL/2305.16739].
49. Fabbri, A.R.; Kryściński, W.; McCann, B.; Xiong, C.; Socher, R.; Radev, D. SummEval: Re-evaluating Summarization Evaluation, 2021, [arXiv:cs.CL/2007.12626].
50. Durmus, E.; He, H.; Diab, M. FEQA: A Question Answering Evaluation Framework for Faithfulness Assessment in Abstractive Summarization. In Proceedings of the Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics; Jurafsky, D.; Chai, J.; Schluter, N.; Tetreault, J., Eds., Online, 2020; pp. 5055–5070. https://doi.org/10.18653/v1/2020.acl-main.454.
51. McKinney, W. Data Structures for Statistical Computing in Python. In Proceedings of the Proceedings of the 9th Python in Science Conference; van der Walt, S.; Millman, J., Eds., 2010, pp. 51 – 56.
52. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* **2011**, *12*, 2825–2830.
53. Hagberg, A.; Swart, P.; S Chult, D. Exploring network structure, dynamics, and function using NetworkX. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 2008.
54. Brin, S.; Page, L. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems* **1998**, *30*, 107–117. Proceedings of the Seventh International World Wide Web Conference, https://doi.org/https://doi.org/10.1016/S0169-7552(98)00110-X.
55. Bird, S.; Klein, E.; Loper, E. *Natural language processing with Python: analyzing text with the natural language toolkit*; " O'Reilly Media, Inc.", 2009.

56. Zhang, T.; Kishore, V.; Wu, F.; Weinberger, K.Q.; Artzi, Y. BERTScore: Evaluating Text Generation with BERT, 2020, [arXiv:cs.CL/1904.09675].
57. Reimers, N.; Gurevych, I. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks, 2019, [arXiv:cs.CL/1908.10084].
58. Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; et al. Transformers: State-of-the-Art Natural Language Processing. In Proceedings of the Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Online, 2020; pp. 38–45.
59. Xu, B.; Chen, Y.; Wen, Z.; Liu, W.; He, B. Evaluating Small Language Models for News Summarization: Implications and Factors Influencing Performance, 2025, [arXiv:cs.CL/2502.00641].
60. Muennighoff, N.; Rush, A.M.; Barak, B.; Scao, T.L.; Piktus, A.; Tazi, N.; Pyysalo, S.; Wolf, T.; Raffel, C. Scaling Data-Constrained Language Models, 2025, [arXiv:cs.CL/2305.16264].
61. Dorfner, F.J.; Dada, A.; Busch, F.; Makowski, M.R.; Han, T.; Truhn, D.; Kleesiek, J.; Sushil, M.; Lammert, J.; Adams, L.C.; et al. Biomedical Large Languages Models Seem not to be Superior to Generalist Models on Unseen Medical Data, 2024, [arXiv:cs.CL/2408.13833].
62. Zhou, Y.; Liu, X.; Zhang, X.; Ning, C.; Wang, S.; Hu, G.; Wu, J. Investigating and Mitigating Catastrophic Forgetting in Medical Knowledge Injection through Internal Knowledge Augmentation Learning. *OpenReview* **2025**. Preprint. Available online: https://openreview.net/forum?id=i9RDDi2SZC.
63. Jin, K.; Wang, Y.; Santos, L.; Fang, T.; Yang, X.; Im, S.K.; Oliveira, H.G. Reasoning or Not? A Comprehensive Evaluation of Reasoning LLMs for Dialogue Summarization, 2025, [arXiv:cs.CL/2507.02145].
64. Irugalbandara, C.; Mahendra, A.; Daynauth, R.; Arachchige, T.K.; Dantanarayana, J.; Flautner, K.; Tang, L.; Kang, Y.; Mars, J. Scaling Down to Scale Up: A Cost-Benefit Analysis of Replacing OpenAI's LLM with Open Source SLMs in Production, 2024, [arXiv:cs.SE/2312.14972].

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.